Timestamp Certificate

This timestamp was created with *Ethereum*

originstamp

Timestamp

Jul-26-2024 06:00:11 UTC

Comment: --- Landscape Printout re. Email to Loudly ---URGENT_ Requesting _License Agreement_ printout from Loudly.com w_ Loudly's letterhead.pdf

Hash:

629aa9c61caee82e152d2018299c7e6e38797a8439875f75457e26ca7a238a54

Transaction:

[0x1797c6e380a738f5d26e2e73d66e99dae42b1e415b9e2358f3d7666b888af9de]

Root Hash:

8fd4518fa649a53a10e25b4eca59580bbb5d339e56722d59f31a78c1a298ade7

Click here to verify your timestamp.

This certificate is only valid in combination with the original file and OriginStamp's open procedure. More information on https://verify.originstamp.com.

Timestamp Certificate

Proof

The proof is necessary for the reproducibility of your document.

```
<?xml version="1.0" encoding="UTF-8"?>
<node type="key" value="8fd4518fa649a53a10e25b4eca59580bbb5d339e56722d59f31a78c1a298ade7">
 <left type="mesh" value="d201a3d4ff548d168ee4e993d0244c1cbb1f2907a72d3610400ebeb85ab05e75">
  <left type="mesh" value="e979e2a2e4e43e7abe7bb7e6ff957f18159ff295abacaaf16d36d9c5b8dce347"/>
  <right type="mesh" value="95b626f117f6036995d00cdcffda6fa557e47e50273386a7a970894a2540f618">
   <left type="mesh" value="284f388184f6927c77db1be6c9259454cb22857f1ab693b352a94e477f1b4552"/>
   <right type="mesh" value="9057e0ff142c5ad1cd9df095d1bfb1e924000fca3fd7de7ea83dc4597a5aa4a6">
    <left type="mesh" value="fa7216613cfe0c3ffc8d89968c091d7421d175e360764989d51d35260dbb7b6b">
     <left type="mesh" value="972c689a9d8e1ffb38dc2382b881158852d0dcef64f79530cd1e4708f4fdbfc4">
      <left type="mesh" value="f2bbaddf2382043998749fdcd09a6b844b8da361a3aaa6112d13ea5152941590">
       <left type="mesh" value="9f85f2c7b658029f9af0684ce08be0d9c9074c81d01d79d8c650055f5980bde5">
        <left type="mesh" value="1f16d054a7fd7205d6af6fdc9ff85e2eb62ef8d03893df12f6fcd2171f5882af">
         <left type="mesh" value="9ed19128f4f4a67b8e8c2415f684f849f4f56a6f44ba8ac075cbc63fea94ac09"/>
         <right type="mesh" value="15c099a2ae4a37cca38bd5d95edb46aa5cfd1581cf5ebd00d56655bd2854c091">
          <left type="mesh" value="3c5c8fab0cd741eebac534bd60c1213ed5d6981ecd6d909577dc624d1803588f">
           <left type="mesh" value="df4253d61c249f5c4047599eef6b86d5426eb44622186ac02212b80c102009de"/>
           <right type="mesh" value="4dfa4c9e38193ce0e7ab48e92324338e51ffadeab0bd44328013e7e0ffc6f19a">
            <left type="mesh" value="14f9fb1c76e38a6ef653fc242766d7bafaab38a73ab6f1ba7efdaa4f7c2b5f7f"/>
            <right type="mesh" value="f62be8488e81a25ccbe81a4ae6ec8b423d07dadb3c9af7ebb9b04a3d74cca6f1">
             <left type="mesh" value="0bf845c78be1a4cfe9462aa011a4fa0b8b5466698a9f617985458fe40183de89">
              <left type="hash" value="629aa9c61caee82e152d2018299c7e6e38797a8439875f75457e26ca7a238a54"/>
              <right type="mesh" value="629cc059fc8eba3546ac8061d304e86c56be22af880b57e0a979b85b8febf511"/>
             </left>
             <right type="mesh" value="d2b25f9b744368e9038b4965212235e0bfb4c7ea9c4c238f41cb8ed5e48f2d98"/>
            </right>
           </right>
          </left>
          <right type="mesh" value="450ce2df9e691520241f75de2cb4ad210ad9a07be1782d5f4753314178bb2488"/>
         </right>
        </left>
        <right type="mesh" value="1f95a88b16ec557aeca13908d2f999d68e5548cfdd596ffaaf31d0ca3e1b4e51"/>
       </left>
       <right type="mesh" value="d5cabb755d03afc456a23a5e6a5abd5ff89afc9a60f8d122fa217335a954ed60"/>
      </left>
      <right type="mesh" value="990b61f6d5b980cffcb537d3bc1068927034725adde685c4688b1e89696c2e78"/>
     </left>
     <right type="mesh" value="e6bd483ecb6cb0ff7ee0038a3272f19f998a9e4b5316a6e30fc88bb7ad37e2e0"/>
    </left>
    <right type="mesh" value="9eb0928a39407133d358bbfa3eae7a1d9b54c6eea0c9b7c5a0f4488cda1985cb"/>
   </right>
  </right>
 </left>
 <right type="mesh" value="0094fedecd5dafa504f4c67268e2d74bde459360227ce44eded6782b674522f8"/>
</node>
```

Timestamp Certificate

Verification

OriginStamp is a timestamp service that uses various blockchains like the Bitcoin Blockchain to create tamper-proof timestamps for your data. Instead of backing up your data, OriginStamp embeds a cryptographic fingerprint in the blockchain. It is de facto impossible to deduce the content of your data from your fingerprint. Therefore, the data remains in your company and is not publicly accessible. All you need to do is send this fingerprint to OriginStamp via the interface. The integration of the RESTful API is very simple and convenient and allows all data to be easily tagged with a tamper-proof timestamp. This document shows the procedure and gives instructions for verifying a timestamp created with OriginStamp.

1. Determine the SHA-256 of your original file

There are numerous programs and libraries to calculate the SHA-256 of a file, such as MD5FILE. Simply drag and drop or select your file, to retrieve the SHA-256 of your file.

2. Validate your proof

First, it must be verified that the hash of the original is part of the evidence. In order to check this, the proof can be opened with a conventional editor and its content can be searched for the hash. If the hash cannot be found, either the file was manipulated or the wrong evidence was selected.

3. Determine the root hash

The Merkle tree is a tree structure, that allows to organize the seed more efficient than a plain-text seed file. The tree is built from the bottom to the top and follows a defined schema. The value of a node is determined by the aggregated hash of its children.

Left child = a8eb9f308b08397df77443697de4959c156fd4c68c489995163285dbd3eedaef

Right child = ab95adaee8eb02219d556082a7f4fb70d19b8000097848112eb85b1d2fca8f67

Node = SHA-256(a8eb9f308b08397df77443697de4959c156fd4c68c489995163285dbd3eedaefab95adaee8eb02219d556082a7f4fb70d19b8000097848112eb85b1d2fca8f67) = 47e47c96302eeba62ed443dd0c89b3411bbddd2c1ff6bdfb1f833fa11e060b85

This step is performed for all levels of the tree until the hash of the root has been calculated. If the hash of the root is identical as proof, the calculation was successful and the root hash is verified. The top hash corresponds to the root hash we embedded in the blockchain through a transaction. For a more detailed explanation of the Merkle tree, we want to refer to Wikipedia.

4. Determine the Ethereum Transaction

Having determined the root hash in the previous step, we store this hash in a Smart Contract in the Ethereum blockchain.

5. Check the transactions

The respective log events must be searched for each transaction in this contract. These events are divided into topics. The root hash should be contained in a topic. Caution: Some services display the hashes with a *0x* prefix. As soon as the transaction has been found, the block time is the actual tamper-proof timestamp. To simplify the search, we added the transaction to the certificate.